UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/822,426	04/12/2004	Paul Deane	128534-02701 (07027570)	2552

26565 7590 11/25/2009
MAYER BROWN LLP
P.O. BOX 2828
CHICAGO, IL 60690

EXAMINER	
GISHNOCK, NIKOLAI A	
ART UNIT	**PAPER NUMBER**
3715	
NOTIFICATION DATE	**DELIVERY MODE**
11/25/2009	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

ipdocket@mayerbrown.com

Office Action Summary	Application No.	Applicant(s)	
	10/822,426	DEANE ET AL.	
	Examiner	Art Unit	
	NIKOLAI A. GISHNOCK	3715	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE _3_ MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on _22 September 2009_.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) _7,10,12-15,20-25,28-30,33,35-37,40 and 41_ is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) _7,10,12-15,20-25,28-30,33,35-37,40 and 41_ is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on _12 April 2004_ is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☐ Notice of References Cited (PTO-892)

2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☐ Information Disclosure Statement(s) (PTO/SB/08) Paper No(s)/Mail Date _____.

4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) ☐ Notice of Informal Patent Application

6) ☐ Other: _____.

DETAILED ACTION

In response to Applicant's remarks filed 9/22/2009, claims 1-6, 8, 9, 11, 16-19, 26, 27, 31, 32, 34, 38, & 39 are cancelled. Claims 7, 10, 12-15, 20-25, 28-30, 33, 35-37, 40, & 41 are pending.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. The factual inquiries set forth in *Graham* **v.** *John Deere Co.*, 383 U.S. 1, 148 USPQ 459 (1966), that are applied for establishing a background for determining obviousness under 35 U.S.C. 103(a) are summarized as follows:

1. Determining the scope and contents of the prior art.
2. Ascertaining the differences between the prior art and the claims at issue.
3. Resolving the level of ordinary skill in the pertinent art.
4. Considering objective evidence present in the application indicating obviousness or nonobviousness.

3. Claims 7, 10, 12, 13, 20-25, 28-30, 33, 40, 42, & 43 are rejected under 35 U.S.C. 103(a) as being unpatentable over Sweitzer, in view of Bloom et al. (US 5,597,312), hereinafter known as Bloom, and further in view of Erickson (US 5,902,114 A), hereinafter known as Erickson.

4. Sweitzer teaches a computer-implemented method of automatically generating a mathematical word problem assessment item, the method comprising: receiving one or more word problem parameters from a user (using an editing window, the user selects the problem type {e.g., multiple choice, free response} and lays out the elements of the word problem with menu and dialog choices, 9:61-63); identifying a plurality of number variables based on the one or more word problem parameters (Variables are used as elements of mathematical

expressions, and as substitution string place holders, 11:22-29); determining a relationship between a first number variable and a second number variable of the number variables (Whenever an instance of one problem is generated {e.g., for preview in the authoring tool or rendering by the print engine}, values are computed for each of the variables according to the variation rules and these values are substituted in the text or in the mathematical expression, 11:22-29).; generating an assessment item using a processor (Whenever an instance of one problem is generated {e.g., for preview in the authoring tool or rendering by the print engine}, values are computed for each of the variables according to the variation rules and these values are substituted in the text or in the mathematical expression, 11:22-29) including: storing the assessment item in a computer-readable memory (11:57-63); [Claims 7, 10, 20-22, & 40].

5. What Sweitzer fails to explicitly teach is wherein generating the text phrase comprises choosing by the processor one or more of word order, word choice, word format, sentence structure, grammar, and language of the text phrase based on the determined relationship [Claims 7, 10, 20-22, & 40]. However, Bloom teaches a method and system for authoring intelligent tutoring, including an exercise function (an exercise function supports trainees working through typical conversations where the goal is to handle some customer request, 6:4-12), using a grammar builder to define situations, actions and other grammar nodes and a sequence and condition of their execution for a tutoring conversation (authoring system and teaching parameters editor themselves comprise conversation knowledge and instructional strategy functions, respectively, designed to support the instructions delivered by tutor. The conversation knowledge function enables instructional designers and domain experts to build knowledge bases of conversations for use by the exercise function of the tutor. The instructional strategies function enables instructional designers to adjust the parameters which the tutor uses to make it tutoring decisions, 5:58-61). The situations and actions are variables defined and

linked by the user (the grammar builder is a graphical tool for building and editing discourse grammars. The author would be presented with another set of graphical tools designed to support the building of situation-action rules embedded within the top level grammar object. This would include tools for defining and linking situations and actions, identifying their type and the type of branch created downstream, and inputting feedback and false responses associated with those rules, 19:36-53). The author of an assessment item having variables, as in Sweitzer, would use Bloom's authoring system to define the relationships between those variables (referred to as parameters by Bloom) and generate the assessment item based on the relationships (in the fashion that the conversations of Bloom are generated), in order to support an item naturally in a language. Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have defined the variables of the word problem generator of Sweitzer using the grammar node relationships as taught by Bloom, in order to train a student using natural language [Claims 7, 10, 20-22, & 40].

6. Sweitzer teaches wherein the assessment item is a mathematical word problem and outputting an assessment item (system for formatting and printing an examination having one or more mathematical expressions, 5:20-29; It is further evident from Figure 3, Items 40 & 42 that the words, "Add two simple fractions without reducing. Add: 'A'/7 + 'B'/14" is a mathematical word problem) [Claims 7, 10, 20-22, & 40].

7. What Sweitzer fails to specifically teach is where generating the mathematical word problem includes generating a text phrase positioned between a first numerical value corresponding to the first number variable and a second numerical value corresponding to the second number variable based on the determined relationship [Claims 7, 10, 20-22, & 40]. However, Erickson teaches a central depository of sample word problems and math facts accessed via the Internet, where a word problem component construction processes is used:

identify a primary component in the word problem with its corresponding derivation, expression, dimension and description; identify a conversion component whose dimension contains a unit contained in the dimension of a previously defined component; identify a compound component that can be constructed from previously defined components; and identify a functional relationship between the components defined above using schematic diagrams and tools from the tool box. The component construction steps end when there exists sufficient components to completely characterize or solve for the variables. In FIG. 3, the components are listed in the order in which they appear in the word problem, and in the order in which they are constructed. (at 5:5-6:17; see also Figures 2-6). It is apparent that the mathematical word problems of Erickson are generated by generating a text phrase, and positioning it between a first and second numerical input value, based on the mathematical characterization of the problem. This specific format of word problems taught by Erickson, would merely be used in the system and method of Sweitzer to match the generated problems to a different standardized test. Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have generating the mathematical word problem of Sweitzer by generating a text phrase positioned between a first numerical value corresponding to the first number variable and a second numerical value corresponding to the second number variable based on the determined relationship, as discussed by Erickson, and in light of the teachings of Bloom, in order to format the assessment items as desired by the standards of the mathematical word problem test [Claims 7, 10, 20-22, & 40].

8. Sweitzer teaches a method of automatically generating an assessment item, the method comprising: receiving one or more inputs pertaining to the format of an assessment item, and determining a format for the assessment item (authoring tool allows {users to} change between answer formats without loss of information already entered, 10:46-51) and selecting one or

more variables for use in the assessment item (variables may be replaced with numbers, text, graphics, or mathematical expressions, 12:30-32); [Claims 10 & 21]; What Sweitzer fails to teach is wherein the format of the assessment item comprises at least one event; wherein each variable is assigned to an event; determining a relationship between variables assigned to an event; and generating an assessment item based on at least the format for the assessment item and the relationship between variables assigned to the at least one event [Claims 10 & 21]. However, Bloom teaches action definitions used in the grammar builder (Action definitions are defined in terms of the following attributes: (i) a comment which is an area for comment by the authors; (ii) a text which is the actual action definition or correct response expected from the trainee; (iii) a type which identifies the type of situation; (iv) a feedback which is a specific feedback to present to the student should their response be incorrect; and (v) a false response which is a specific response to be used as distracters when presenting the trainee with multiple choice options. The text attribute can be in the form of text spoken by an expert CSR, the correct answer textually presented to the trainee or specific application commands, again depending upon the identifier in the type attribute, 19:6-14), which are linked in a relationship by the author using the grammar builder (19:48-53). The situation definitions are understood to be actions, as they define situations presented to a user in natural language for the user to solve. The situation definitions are defined terms of attributes (18:65-67). The attributes {parameters} of Bloom are understood to be variables assigned to the event; the assessment items {conversations} are generated based on the relationships between the variables {such as the correct response expected or distracters}. The grammar builder is understood to define the format of the assessment item {conversation}, e.g. a syntax which defines the specific situations, actions, and other grammar nodes, and a sequence and/or conditions of their execution, 18:61-64). The variables of Sweitzer, used to generate mathematical word problems

as in Sweitzer, would be formatted into natural language events by the grammar builder of Bloom, where the variables as in Bloom are assigned to the events, and the item is generated based on the relationships between the variables, as in Bloom, for the purpose of defining a situation or action definition {an event} in natural language for a user to clearly understand the problem. Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have defined the word problems generated as taught by Sweitzer, using the events and their relationships as taught by Bloom, in light of the teachings of Erickson, in order to train a student using natural language [Claims 10 & 21].

9. Sweitzer teaches wherein determining a relationship for the variables assigned to each event comprises one or more of the following: determining a variable for which to solve for each event ("WHAT IS THE TOTAL PRICE OF 'APPLES' APPLES IF THE PRICE OF EACH IS $'PRICE' ?"; “KEY: $’TOTAL’", Figure 5) and determining an answer for each event ("WHAT IS THE TOTAL PRICE OF 2 APPLES IF THE PRICE OF EACH IS $0.20 ?"; “KEY: $0.40”, Figure 5); determining a value for one or more variables (Variables may be replaced with numbers, text, graphics, or mathematical expressions, 12:30-32); and determining a variable format (The authoring tool allows problems to be changed between answer formats without loss of information already entered. Users may select a multiple choice format with two to five candidate answers or distracters, one of which is correct; a true/false format; or a free response format, where an exam taker writes an answer long hand. In the multiple choice format, distracters have diagnosis information attached, 10:46-51) [Claim 12].

10. Sweitzer teaches wherein determining a format for the assessment item comprises: determining a problem format having one or more sections (multiple-choice format, 10:46-53; the plurality of choices are understood to be sections); and determining content to place within each section (the variation rules engine processes the variation rules from a {multiple-choice}

problem from the top down, and each line specifies a calculation, constraint, or an external function call, 17:43-51) [Claim 13].

11. Sweitzer teaches a method of automatically generating an assessment item, the method comprising: receiving one or more input parameters (The authoring tool and print engine, discussed in greater detail below, use variation rules which are stored in the variation rules module or engine and define instances of a generalized problem. The variation rules associated with a given problem guide the replacement of variables incorporated in each problem, 12:30-32); generating a document structure based on the one or more input parameters (The print engine reads worksheets and problem books. Further, the print engine sequences through multiple students; question, answer, and key sheets for each student; and the problems on each sheet. It then evaluates the random variations in the problem, stores key values for later use in scoring, and arranges the problems and annotations on an output, which is either a printed sheet or a visual display on the video monitor, 18:33-40; inputs are the random variables in the problems) [Claims 22 & 40]. What Sweitzer fails to teach is producing a logical schema using the document structure; and generating an assessment item based on the logical schema [Claims 22 & 40]. However, Applicant defines a “schematic logical representation” as information contained in a file pertaining to the syntax and vocabulary used to generate natural language for a given assessment item, defining one or more variables for which text may be generated (Page 8, Para. 0038). Bloom’s conversations are taught to be syntactically-correct sequences of situations and actions, which form a variable path through the discourse. Bloom also explains that the conversations help to overcome the limitations of natural language understanding (8:45-56); in that a user selects a naturally organized response they can understand rather than a symbolic response that may be misinterpreted. The conversation mapping of Bloom is understood to represent a logical schema, having information {commands,

text, audio, branches, and rules} stored in a file {reusable nodes in a course} pertaining to the syntax and vocabulary {syntactically-correct conversation sequences} and defining variables for which the text may be generated {branching in the conversation is based on customer information} used to generate natural language (Conversations are syntactically correct sequences through discourse grammar made up of sequences of situation-action rules. Each abstract situation and action in a conversation sequence is instantiated with specific information in the form of application commands or information, or text with accompanying audio. Conversations are grouped together to reflect different types of scenarios that could occur between a caller and a CSR. Branches within conversations are based on customer information. Situation-action rules that are conceptually related map onto discourse grammar nodes. These nodes are reusable portions of conversations that can appear in several different conversation scenarios. A course has a title, list of topics, a list of grammars, a list of activities (situations or actions), a list of conversations (subsuming text, audio, and application communication) and a list of application specifications; a grammar is a set of conversations in an AND/OR tree, where nodes are situation-action pairs, and branches are different possibilities based on the customer situation, all at 15:18-62). The logical schema of Bloom would be used to format the assessment items of Sweitzer in natural language. Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have used the conversation generator as taught by Bloom, in the problem generator of Sweitzer, in light of the teachings of Erickson, in order to generate natural language problems [Claims 22 & 40].

12. What Sweitzer further fails to explicitly teach is wherein generating a document structure comprises: building a mental model; and outlining the document structure based on the mental model [Claim 23], and wherein building a mental model comprises: selecting one or more semantic frames; generating a list of one or more events; and binding one or more variables

across the one or more events [Claim 24], and wherein outlining the document structure comprises: generating one or more sentences for the mental model; determining a function for each sentence; and determining information to express in each sentence [Claim 25]. However, Applicant defines a “mental model” as a structure with a limited number of component entities and relationships, varying along some dimensions (Page 9, Para. 0040), and a “semantic frame” as a method of organizing verbal content, making it possible to analyze a type of problem and isolate a series of variables (Page 8, Para. 0036). Bloom teaches a student model (student model is essentially a dynamic data structure that maintains a "model" of each trainee within and across training sessions. Student model is used to actively assess the trainee's mastery of the learning materials, represent the trainee's progress through the learning materials and recommend topics or conversations the trainee needs to study or practice, select and present instructional interventions at appropriate levels of understanding, apply selected instructional strategies during contact rehearsals, and provide the trainee with performance feedback; 11:25-12:65), which outlines the topics taught to the student, and represents all the tasks and conversations available to the student within each topic (student model represents contact scenarios in a hierarchical network, with scenarios in major topics at the highest level, and individual conversation activities at the lowest level. Action can be verbal (i.e., responses by the CSR to the customer or to service order software output), operational (i.e., commands or data entered into the service order software by the CSR), or cognitive (i.e., information gathering, information processing and decision making). The top level is referred to as the scenario or topic node level and represents all of the possible conversational scenarios in which the CSR can engage. The second level is referred to as the task or sub-topic node level and represents all of the major tasks or parts of conversations within each scenario or topic. The lowest level, previously described, is referred to as the action node level and represents the individual

actions available to the trainee, 11:52-12:4; see also 19:65-20:29). The student model of Bloom is understood to represent a mental model, in the processes of building the mental model {representing the trainee's progress through the learning materials and making representations of topics the trainee needs to practice}; selecting semantic frames, *as defined by Applicant* {the hierarchical nature of the student model is organized verbal content; the branching nodes of the topic, conversation, and action levels allow selection of variables for a given problem type}; generating a list of events {the verbal, operational, or cognitive actions available to a student}; binding variables across events {the event variables are those stored in the data structure of the student model; application descriptions include information about input and output fields and the values expected in them, 16:9-13}; and outlining the document structure based on the mental model {the node levels represent all the parts of the conversations used to generate a dialog}. Thus the conversations of Bloom are generated using the student model, and the conversations used to build a document structure using the user interface (Student model comprises a dynamic representation of the student's state of knowledge. The user interface comprises a communication channel between the tutor {software} and the student, 4:17-20). The student model of Bloom, containing the progress of the student as well as the variables and relationships used to generate assessment conversations by the tutor software, would be used to generate the mathematical word problems in the assessment item generator of Sweitzer. Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have used the mental models as described by Bloom, in the word problem generator of Sweitzer, in light of the teachings of Erickson, in order to reflect natural language in the word problem [Claims 23-25].

13. Sweitzer teaches storing the document structure in a file (The authoring tool stores problem descriptions in files, 9:52-53). What Sweitzer, Bloom, and Erickson fail to explicitly

teach wherein the document structure file is an XML file. However, Applicant has not disclosed that having the file be an XML file solves any stated problem or is for any particular purpose. Moreover, it appears that the Microsoft Word type files of Sweitzer (3:52-61) or the Applicant's instant invention would perform equally well for storing a variable document specification. Accordingly, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have modified Sweitzer, Bloom, and Erickson to use an XML file, because such a modification would have been considered a mere design consideration, which fails to patentably distinguish over Sweitzer, Bloom, and Erickson.

14. Sweitzer teaches wherein the document structure includes an outline of a sequence of one or more sentences for the assessment item (The authoring tool and print engine use variation rules which are stored in the variation rules module or engine and define instances of a generalized problem. The variation rules associated with a given problem guide the replacement of variables incorporated in each problem. Variables may be replaced with numbers, text, graphics, or mathematical expressions. The variation rules for a problem are an ordered list of definitions and constraints expressed in a simple language. The variation rules may assign substitution variables (variables used in the problem layout) or temporary variables (variables used only within the variation rules). The variation rules may also impose constraints on the relationship between variables, 12:25-67; definitions and constraints are in a simple language, understood to be simple sentences; also, dynamic problems are varied using sequencing and randomization, and the sequencing and randomization functions keep historical information to avoid problem repetition, 15:48-65) [Claim 28].

15. What Sweitzer further fails to explicitly teach is wherein producing a logical schema comprises: outlining a sentence structure for one or more sentences; and determining an information format for each sentence [Claim 29], wherein determining an information format

comprises determining an ordering of one or more elements for each sentence [Claim 30]. However, Bloom teaches wherein producing a logical schema comprises: outlining a sentence structure for one or more sentences (situation-action rules map onto discourse grammar nodes, the nodes are reusable portions of conversations that can appear in different scenarios, 15:54-57); and determining an information format for each sentence (Discourse grammar defines abstractly a set of possible discourses using an RTN {a recursive transition network; defining the structure of a grammar}. RTN formalism permits a discourse to be decomposed into grammars that describe specific sub-discourses for readability and reuse. Discourse grammar is a collection of grammars of which one is the top level grammar. An individual grammar definition preferably includes a name, a nickname, author comments about the grammar, a context describing the purpose of this part of the discourse to a student, a list of topics that this grammar covers, a syntax defining the paths, the list of activities used by the grammar, the grammars that this grammar calls, the grammars called by this grammar, and the depth of the grammar within the grammar calling structure. The RTN permits situation-action pairs to be strung together as a sequence of steps to take with possible decision points in the sequence. A Backus-Naur Form (BNF) specification is the syntax for defining the transition networks, 16:41-58; BNF notation is understood to provide an outline of rules for sentence structure and defining a format of a language, either in English or another natural language as well as a computer programming language). The logical schema {conversation mapping} using situation-action rules {the outline of sentence structure} and BNF notation rules {the information format} of Bloom would be used to generate natural language in the assessment item generator of Sweitzer, in order to abstractly define a set of possible discourses for readability and reuse. Bloom also teaches wherein determining an information format comprises determining an ordering of one or more elements for each sentence (the situation-action nodes of grammar

form a path, 15:44-62). The mapping situation/action rules of the discourse grammar nodes of Bloom impart a logical order of a conversation, which the BNF notation rules of Bloom define an order to individual sentences. The rules of Bloom would be used to generate an assessment item in Sweitzer that improves readability and reuse of the item. Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have produced a logical schema for use in the assessment item generator of Sweitzer, by outlining a sentence structure for one or more sentences, and determining an information format for each sentence, wherein determining an information format comprises determining an ordering of one or more elements for each sentence, as taught by Bloom, in light of the teachings of Erickson, in order to abstractly define assessment item language for improved readability and reuse [Claims 29 & 30].

16. What Sweitzer further fails to teach is storing the logical schema in a file. However, Bloom teaches storing the logical schema in a file (each piece of multimedia is contained in separate files, 7:1-10). The logical schema stored in a file of Bloom would be used to generate an assessment item in the invention of Sweitzer, in order to format the assessment items of Sweitzer in natural language, using a reusable file. Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have used the conversation generator as taught by Bloom, in the word problem generator of Sweitzer, in order to generate word problems that follow natural language, improving the reusability of the logical schemas by storing them in files. What Sweitzer and Bloom fail to explicitly teach is wherein the logical schema file is an XML file. However, Applicant has not disclosed that having the file be an XML file solves any stated problem or is for any particular purpose. Moreover, it appears that the Microsoft Word type files of Sweitzer (3:52-61) or the Applicant's instant invention would perform equally well for storing a variable document specification. Accordingly, it would have

been obvious to one of ordinary skill in the art, at the time the invention was made, to have modified Sweitzer, Bloom, and Erickson to use an XML file, because such a modification would have been considered a mere design consideration, which fails to patentably distinguish over Sweitzer, Bloom, and Erickson.

17. What Sweitzer further fails to teach is wherein generating an assessment item comprises: parsing the logical schema; annotating the parsed logical schema with grammatical information; determining words and word forms based on the grammatical information; and outputting text representing the assessment item [Claim 33]. However, Bloom teaches wherein generating an assessment item comprises: parsing the logical schema (The conversation author is a tool that allows authors to create conversations based on specific paths through or parts of the developed discourse grammar. The conversation author works by having the author select the grammar path, or part, to be instantiated. Next, they either select an existing conversation to edit, or else create and name a new conversation. Once the conversation is identified, they then execute the author functions, 19:60-62; designing and traversing the paths of the discourse grammar is understood to be parsing; also, parsing is also used to describe decoding a grammar using the Backus-Naur Forms); annotating the parsed logical schema with grammatical information (During authoring, the method and system of the present invention execute the selected grammar path, or part, and present the author with a variety of different input fields depending upon the specific situation or action being executed. In the case of "verbal situations", the author would type in the customer statement, request or question into the verbal situation input field. In the case of "operational situations", the author would type in the simulation situation input field, the name of the screen and field that the resulting action would take place in. In the case of implicit situations for which there is no overt clue, or situations in which information is to be given to the trainee unconditionally {cognitive actions}, the author

would input the information in the situation input field, 19:66-20:3; the author's input of verbal, operational, or cognitive actions into the input fields is understood to be annotating the schema with grammatical information); determining words and word forms based on the grammatical information (In the case of verbal actions, the author would type in the correct response in the verbal action field, 20:13-15; a correct response would inherently be a grammatically correct response, else the author caused the response to be incorrect); and outputting text representing the assessment item (Once the conversation is complete, it is saved and automatically indexed according to its high level scenario, as well as by the specific sub-topics of conversation components that comprise it. The saved conversation can then be edited or else used by the system and method of the present invention, 20:23-29; saving is understood to be outputting the assessment item conversation to disk; using the conversation is understood to by outputting it to a user). The specific steps of generating the assessment item using a logical schema, as taught by Bloom, would be used in the system and method of Sweitzer for generating a mathematical word problem, in order to provide authoring tools and automatic methods or a non-programmer instructional designer to easily generate the knowledge bases used to create the assessment items. Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have used the method of Bloom to generate an assessment item, comprising parsing the logical schema; annotating the parsed logical schema with grammatical information; determining words and word forms based on the grammatical information; and outputting text representing the assessment item, in the method for generating an assessment item taught by Sweitzer, in light of the teachings of Erickson, in order to provide authoring tools and automatic methods or a non-programmer instructional designer to easily generate the knowledge bases used to create the assessment items [Claim 33].

18. Sweitzer and Bloom teach all the features as demonstrated above in claim 10, including wherein the one or more inputs comprise one or more of: a number of events; a number of distinct frames; a type of participant in each event; and a type of assessment item to generate [Claim 40].

19. Sweitzer teaches wherein the first and the second numerical values are one of the identified variables (the "random ()" external function varies the number of apples and their price, so that each student receives a unique question. Calculations are performed as in any common programming language meaning that the result of each calculation is stored in a variable, 17:43-63) [Claim 42].

20. Sweitzer teaches wherein the first numerical value or the second numerical value is a constant identified by the one or more word problem parameters (Mathematical operations use the integer or floating-point elements, according to the governing representation. In the function descriptions shown, the formal input arguments {A, B, etc.} represent expressions. These expressions may include arithmetic operations, variables, constants, or other function calls, 16:32-46; see also Table 11) [Claim 43].

21. Claims 14, 15, 35-37, & 41 are rejected under 35 U.S.C. 103(a) as being unpatentable over Sweitzer and Bloom, and further in view of Wen.

22. Sweitzer and Bloom teach all the features as demonstrated above in claims 10 and 22. Bloom teaches wherein generating an assessment item comprises one or more of the following: selecting a sentence structure for each sentence in the assessment item (6:4-12, 15:54-57, and 19:48-53); selecting identification types for one or more of the variables; determining a numerical format for each of the one or more variables; and determining a verb tense to use for each event [Claim 14]; a method of automatically generating an assessment item, the method

comprising: assigning one or more mental model structure variables (a student model, 11:25-12:65); defining one or more identity variables for a mental model structure (application descriptions include information about input and output fields and the values expected in them, 16:9-13); determining a task-relevant problem structure (generating a list of events by traversing the nodes of the conversation tree, and binding the variables across the events, 16:9-13); defining a document format (conversations used to build a document structure using the user interface (4:17-20); and determining language variations (hierarchical patterns of vocabulary, 15:63-16:2) [Claims 35 & 41], including selecting a referent identification type for each of one or more participants (19:66-20:3) [Claim 41], wherein assigning one or more mental model structure variables comprises defining one or more of the following: one or more events (16:9-13); one or more distinct semantic frames; one or more participant types; and an event type for each event [Claim 36], wherein determining a task-relevant problem structure comprises: determining a variable for which to solve for each event (application descriptions include information about input and output fields and the values expected in them, 16:9-13); determining an answer for each event (10:46-51); and determining one or more values for each variable (The screen item describes an item of the screen and has a name by which it can be referred, a position and a size within the screen, and a default value. The value can be defined as a string or dynamically by a function, 17:31-37) [Claim 37], wherein determining language variations comprises: selecting a sentence structure for each of one or more sentences; selecting a referent identification type for each of one or more participants; and determining a tense for each of one or more events (17:48-18:7).

23. What Sweitzer and Bloom fail to teach is selecting a language for the assessment item [Claim 14]. However, Wen teaches a method and system of teaching a language, using a grammatical engine which collects the grammar of a language {such as English} to provide

grammatical rules (1:55-60). The grammatical engine of Wen identifies a language used to generate an assessment item, based on the grammatical rules, which would merely be a variable in the logical schema of Bloom, defining a particular set of Backus-Naur forms which parse a given language and define the natural language conversations taught by Bloom, used when generating the assessment item taught by Sweitzer. Therefore, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have used the grammar database identifying a language such as English, in Sweitzer's problem generator, in order to teach word problems while teaching a language [Claims 14, 15, 35-37, & 41].

24. What Sweitzer, Bloom, and Wen further fail to teach is wherein selecting identification types comprise determining to identify a variable denoting a person by using a proper name, wherein selecting identification types comprise determining to identify a variable denoting a person generically, wherein selecting identification types comprise determining to identify a variable denoting an object by using a label, wherein selecting identification types comprise determining to identify a variable denoting an object by using a description of the object [Claim 15]. However, Applicant has not disclosed that having proper names, generic names, labels and descriptions for persons or objects solves any stated problem or is for any particular purpose. Moreover, it appears that the variables of Sweitzer or the Applicant's instant invention would perform equally well for selecting identification types. Accordingly, it would have been obvious to one of ordinary skill in the art, at the time the invention was made, to have modified the variable id's of Sweitzer, in combination with Bloom and Wen such that proper names, generic names, labels and descriptions for persons or objects identified variables, because such a modification would have been considered a mere design consideration, which fails to patentably distinguish over Sweitzer, Bloom, and Wen [Claim 15].

Response to Arguments

25. Applicant's arguments with respect to claims 7, 10, 20-22, 35, & 40-43 have been considered but are moot in view of the new ground(s) of rejection.

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to NIKOLAI A. GISHNOCK whose telephone number is (571)272-1420. The examiner can normally be reached on M-F 11:00a-7:30p EST (8:00a-4:30p PST).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Xuan M. Thai can be reached on 571-272-7147. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

11/21/2009
/N. A. G./
Examiner, Art Unit 3715

/XUAN M. THAI/
Supervisory Patent Examiner, Art Unit 3715